

Report No. 7165

Monitoring CAP8 and SURAP Networks (SRNTN-12)

Michael Cote

Prepared By:

BBN Systems and Technologies Corporation
10 Moulton Street
Cambridge, MA 02138

Prepared for:

DARPA/ISTO
1400 Wilson Bl.
Arlington, VA. 22209

Sponsored by:

The Defense Advanced Research Projects Agency
Under Contract No. MDA-903-83-C-0173

The views and conclusions contained in this document are those of the authors and do not represent the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency, the Army or the United States Government.

90 000511

AD-A224 992

Contents

1 Introduction and Summary	2
1.1 NetMan Considerations	2
2 Monitoring Data and Its Uses	3
3 THE MDP PROTOCOL	5
3.1 Justification	5
3.2 MDP Header	5
3.3 MDP Sessions	5
3.4 Session Aborts	6
3.5 LPR algorithms	6
A MDP Packet Formats	8
A.1 The MDP Packet Header	9
A.1 MDP Type # 0 - Radio Request/Command Packet	10
A.2 MDP Type #1 - General Status	13
A.3 MDP Type # 2 - Routing Information	19
A.4 MDP Type # 3 - CUMSTAT Values	20
A.5 MDP Type # 4 - Neighbor LPR Data	22
A.6 MDP Type # 5 - LPR Module Versions	26
A.7 MDP Type # 6 - LPR Fault Queue	27
A.8 MDP Type # 7 - LPR Timer Values	28
A.9 MDP Type # 8 - LPR Command NACK	29
B MDP Triggers	30
C Algorithms for Packet-Length Statistics	31

Accession For	
NTIS CFARI	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
<i>per DTIC</i>	
By	
Date	
Available from	
Distribution	
<i>A-1</i>	

1. Introduction and Summary

Network monitoring is necessary for debugging protocols, measuring network performance, determining routes, and predicting network functionality. These tasks require data collection. Network Monitoring systems must poll Low-Cost Packet Radios (LPRs) for specific types of data, then process it for an operator. The Monitoring Data Packets (MDPs) specified in this document provide a flexible mechanism for monitoring systems to collect this data from an LPR network. Specific SURAP network monitoring applications that require MDPs are: Siting Aids for deploying the LPRs, NetMan for global network management, and the NIU (Network Interface Unit) for local network information.

This document specifies the MDP formats for SURAP1 and SURAP2. These formats (see Appendix A) are the result of merging the packet radio PDP (Performance Data Packet) and the DIP (Device Information Packet). The remainder of the data is derived from the CAP LROP and from information which, in the CAP8 networks, has been available only from the PR itself using the XRAY tool.

1.1 NetMan Considerations

The following section is a brief overview of the SURAP NetMan. For a more detailed description of the various services that the NetMan will provide, refer to SRNTN 26. For the technical design of NetMan, refer to SRNTN 10.

2. Monitoring Data and Its Uses

The data that we gather allows NetMan to provide the operator with the following four basic services:

- views of the current network state – visual depictions derived from the currently available network data which has been reduced to a form more easily interpreted by the operator.
- alerts to critical network events – pointers to possible problems in network behavior to allow for operator follow-up
- tools to load SURAP software
- tools to tune and debug the network.

To display the connectivity, traffic flows, and routes in a particular area, the Network Manager needs to gather data in the following categories:

- component connectivity (PR, CH, SCH)
- component membership (PRs in a cluster, etc.)
- traffic on logical or physical links
- available services – network interface units, node trackers, etc.
- status of components
(up/down/flaky/saturated/reliable/...)
- fault reports.

The Network Manager reduces the large amount of data produced by the various reporting network components into views which the operator can easily interpret and act upon.

The Network Manager design paper, SRNTN 10, lists nine problems which arise in CAP8 and SURAP radio networks. The Network Manager must be able to display the data it has gathered in a way that allows the operator to notice if one of these situations has occurred. The problems listed include:

1. packet radio/clusterhead failure/malfunction
2. excessive traffic forwarding in a cluster
3. excessive control traffic in a cluster
4. labeling/clustering instability
5. failure to find a route
6. failure to find a device
7. route/link failure/fluctuation
8. link/packet radio saturation
9. network resource hogging by a packet radio host.

Ongoing research and continued evolution of the Network Manager is uncovering additional areas in which debugging aids can be provided. These areas include:

1. finding bugs and inefficiencies in network algorithms
2. verifying that these algorithms are correctly implemented
3. validating the algorithm designs
4. facilitating the control of network components
5. analyzing performance
6. supporting administrative needs
7. isolating faults.

The Network Manager must be able to display a spectrum of views ranging from coarse to fine, terse to verbose. The operator can use these views to focus on the relevant data and to narrow down the possible causes of the problem. These views are intended to help the operator examine the three network attributes mentioned earlier — connectivity, traffic flows, and routes . . . , and to allow the operator as much flexibility as possible to aid the operator in debugging problems in the network and in testing network reactions to various operations.

3. THE MDP PROTOCOL

3.1 Justification

Because MDPs use SNAP as the transport protocol, which does not allow for multi-packet messages (and therefore allows for no message fragmentation or reconstruction), a simple MDP protocol has been created. Other aspects of the MDP protocol deal with requesting MDPs from LPRs and with the LPRs' responses to these requests.

3.2 MDP Header

The MDP header serves three functions. First, it allows a monitor to parse the text of the MDP. Second, it timestamps the message. Third, it provides packet sequencing in a multi-packet MDP message.

The version number is used to differentiate between protocol versions. As new protocols are developed, the version numbers are incremented. When a monitor receives an MDP packet with the wrong version number it can prevent itself from incorrectly parsing the packet.

The **MDP (message) type** field is used to determine which of the packet text formats to use to parse the MDP text. Up to 64 types are currently allowed.

The **PR Timestamp** field is a 32-bit segment of the LPR protocol clock's 48-bit number. The least significant eight bits and the most significant eight bits are masked off. Because the LPR's protocol clock has units of 26 milliseconds, the units of the MDP timestamp are 6.656 milliseconds (26 milliseconds \cdot 256 [$256 = 2^8$]). Because the timestamp has 32 bits, it will not wrap around for over 330 days.

For message sequencing, two fields are used, **pkt number** and **# of pkts**. The first is the position of this packet in the message, the second is the total number of packets in a message. A one-packet message would have 1,1 respectively. Packet 2 of a five-packet message would have 2,5 respectively.

3.3 MDP Sessions

To start an MDP session, a monitor transmits a Radio Command/Request Packet (RRCP - MDP Type #0) to the LPR. This packet is used to request the non-event-driven MDPs 2 through 7, to make a one-time request of MDP #1, or to request MDP #1 on an event-driver basis. It is also used to modify the triggers for MDP #1 transmission, or to clear cumulative statistics, also known as CUMSTATs.

The RRCP header contains a version and command type. To request MDP message types 2 - 7, or to make a one-time request of MDP #1, the monitor sets the **sub_type** field to **SEND_MDP** and sets the **MDP type** to the number corresponding to the packet type desired. The monitor should receive one message in response to this request. However, if the LPR is currently servicing a request for some other monitor, the MDP request will be dropped (but SNAP ACKed) and the monitor will receive Radio

Command NAK Packet (RCNP - MDP Type #8) in response; the monitor must retransmit the request at a later time.

To request MDP #1 on an event-driven basis, the monitor sets the `sub_type` field to `SEND_ME_MDP1`. Optionally, the monitor can set the minimum and maximum timers for the MDP #1 timeouts (within certain constraints). The minimum timer sets the minimum number of seconds which must elapse between MDP #1 transmissions. However, the LPR will enforce a thirty-second minimum. The maximum timer sets the maximum number of seconds between transmissions. The last `SEND_ME_MDP1` which specifies the minimum/maximum timers will determine the value of these timers in the LPR. Values of zero for either the minimum or maximum timers will result in the current timer values remaining unchanged. The default value for the minimum timer is thirty seconds; for the maximum timer, the value is thirty minutes (1800 seconds).

As the triggers occur in the LPR, the monitor will receive MDP #1 packets. To request a certain trigger, the monitor must set the bit corresponding to this trigger in the trigger mask field. If the LPR acknowledges the request but an MDP #1 packet has just been transmitted to some other monitor, the LPR will wait the minimum inter-transmission time before transmitting another MDP #1 packet. Otherwise, if it acknowledges the request, it will transmit the MDP #1 packet immediately. If the LPR "NAK"s the request, the monitor will receive an RCNP with the "NAK reason" set to `CANT_SET_MDP_DEST`, and the list of monitors which are already monitoring the LPR.

To cancel event-driven MDP #1 packet transmissions, the monitor sends a `CANCEL_MDP1` request. Successful transmission (i.e., receiving the SNAP ACK) implicitly acknowledges this packet.

To clear the CUMSTATs, the monitor sets the `sub_type` field to `CLEAR_CUMSTAT` and sets the text to a 48-bit number corresponding to the time (in 6.656 millisecond units) at which the LPR should clear the cumulative statistics. This 48-bit number is relative to the timestamp which the LPR has returned in the MDP header. A value of zero indicates that the LPR must clear the cumulative statistics upon receiving the packet.

3.4 Session Aborts

If a monitor has requested an MDP message and knows (since it has received the first packet of the message) that the message is a multi-packet message, the monitor should abort if the next packet does not arrive promptly. However, since the LPR may miss some of the acknowledgements from the SNAP transport and therefore retransmit, two SNAP timeout periods should be the minimum timeout: one for the packet just received and one for the next packet in the sequence. This ensures that messages will not be aborted needlessly.

3.5 LPR algorithms

The LPR must maintain a list of monitors and their corresponding trigger masks. In addition, the LPR requires minimum and maximum timers, a global trigger mask (in which to collect the outstanding triggers), and a "snap-shot" buffer to maintain a copy of an MDP for either multi-destination or multi-packet transmission. After SURAP protocol startup, the LPR will clear the monitoring list; it will transmit no MDPs on an event-driven basis until it receives a request to do so.

As the triggers occur in the LPR, they are collected in the global trigger mask until an MDP #1 packet transmission is again allowed; the time between MDP #1 transmissions cannot be less than the minimum inter-transmission time. Each time the maximum inter-transmission time elapses, the LPR will send an MDP #1 packet to all monitors that have set the timeout bit in their trigger masks. At transmission time, the LPR constructs an MDP #1 packet; the same packet will be sent to all the monitors. If a destination's trigger mask and the global trigger mask have at least one bit in common (logical AND is unequal to 0), the LPR transmits the MDP #1 packet to that destination. The packet is transmitted to an attached monitor first (if there is one); the order of transmissions to the other attached and remote monitors is random.

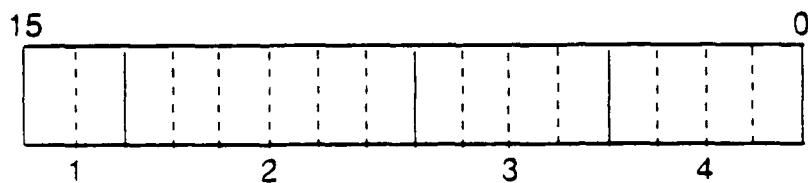
If a request is received while the snap-shot buffer is already in use, the request is acknowledged by SNAP, but the MDP request is dropped and an RCNP is sent to **NACK** the request. The monitor must time out and request the MDP again. No further action is taken by the LPR.

Appendix A. MDP Packet Formats

This Appendix specifies the formats of MDP packet types 0 through 8. Every MDP packet is a sequence of adjacent 8-bit bytes that are organized into fields. Each field is an integer number of bytes that have special significance for higher-level protocols. We present a top-down view of all nine MDP packet types by breaking each into its constituent fields and explaining the meaning and purpose of each.

A.1 The MDP Packet Header

The MDP Packet Header consists of 48 bits. The most significant 16 bits give the version number, MDP type, packet number, and the number of packets in the MDP (Figure 1.).



1. Header version (currently 0)
2. MDP type (0 - 8)
3. Packet sequence number
4. Number of packets in MDP

Figure A.1: MDP Packet Header, first 16 bits

The least significant 32 bits of the header is the MDP creation time (in units of 6.656 milliseconds) given by bits 7 through 39 of LPR.currenttime variable in the LPR (Figure 2.).

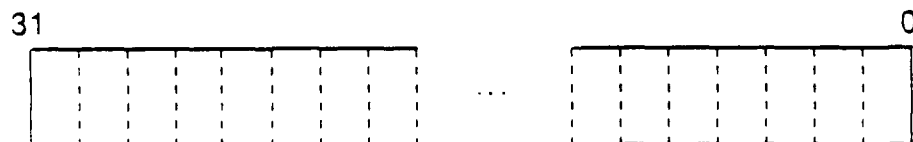


Figure A.2: MDP Packet Header, last 32 bits

A.1 MDP Type # 0 - Radio Request/Command Packet

This packet is used to request that an LPR perform network-control tasks; e.g., send an MDP packet, clear its CUMSTATs, etc. This packet contains the two fields:

1. The header

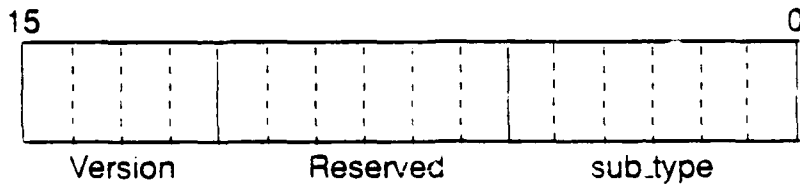


Figure A.3: The MDP Type 0 Header Format

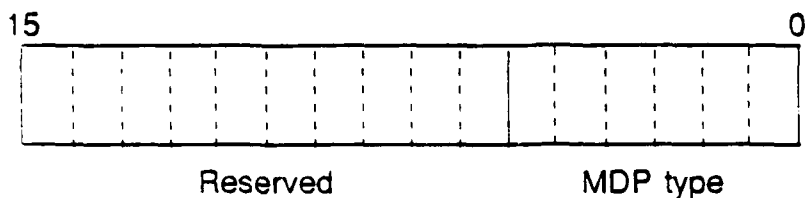
The allowable sub_type field values are:

- (a) SEND_MDP
- (b) SEND_ME_MDP1
- (c) CLEAR_CUMSTAT
- (d) CANCEL_MDP1

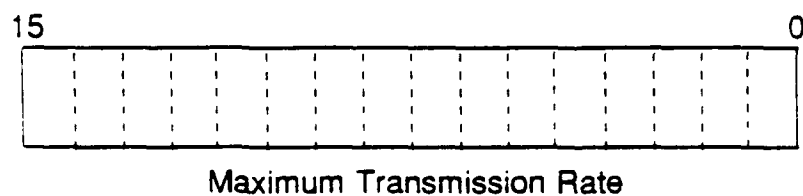
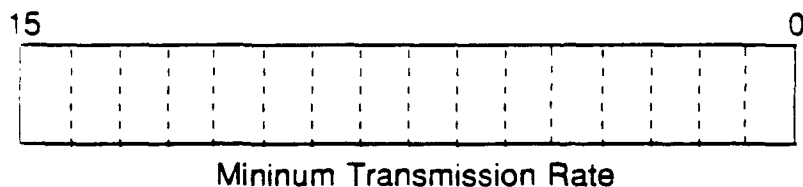
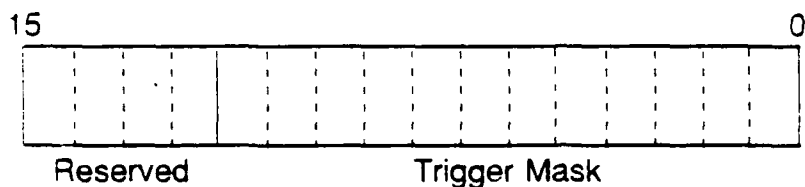
2. The 2-byte data fields

After the packet header there are a variable number of 2-byte data fields.

- (a) If the sub_type field is set to SEND_MDP the 2-byte data fields have the following special format:



- (b) If the sub_type field is set to SEND_ME_MDP.1 the data field has the following format:



Trigger Mask These 12 bits indicate which conditions will trigger the transmission of MDP #1. The conditions are codified as trigger values. (See Appendix B.)

Minimum Inter-transmission Time The minimum number of seconds between successive MDP #1 transmissions. A value of zero means use the current value, or if it has never been set by the Network Manager, use the default value of 30 seconds.

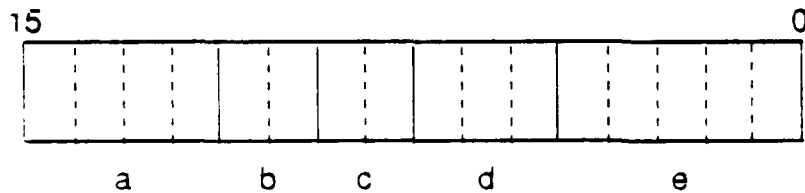
Maximum Inter-transmission Time The maximum number of seconds between successive MDP #1 transmissions. A value of zero means use the current value, or if it has never been set by the Network Manager, use the default value of 1800 seconds.

- (c) If the subtype field is set to CANCELMPD1 the packet has no data field.
- (d) If the subtype field is set to CLEAR_CUMSTAT the data field is 32-bits long. It specifies a protocol time (in units of 6.656 milliseconds) when the LPR is to clear all its CUMSTATs. The value zero indicates causes the LPR to clear its CUMSTATs upon receipt of the packet.

A.2 MDP Type #1 - General Status

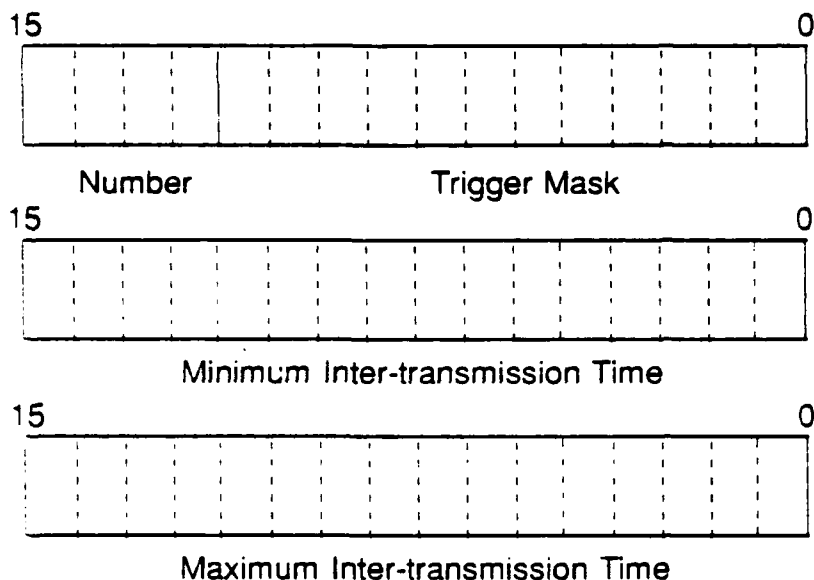
This packet contains LPR status information. It contains the following fields:

1. The header



- (a) MDP version number
- (b) Reserved
- (c) Number of monitors in the packets (1, 2, or 3)
- (d) Number of attached devices
- (e) Number of neighbor LPRs

2. Trigger masks and transmission timers.



Number The number of triggers that are set. This information is used in parsing the trigger counter section. (This field is at the end of the packet.)

Trigger Mask A mask of triggers which have occurred since the last transmission of an unsolicited MDP #1. It is zero for solicited (polled) MPD #1 packets. See Appendix B for a list of triggers.

Minimum Inter-transmission Time The minimum number of seconds between successive MDP #1 transmissions. A value of zero means use the current value, or if it has never been set by the Network Manager, use the default value of 30 seconds.

Maximum Inter-transmission Time The maximum number of seconds between successive MDP #1 transmissions. A value of zero means use the current value, or if it has never been set by the Network Manager, use the default value of 1800 seconds.

3. The next three 32-bit fields contain LPR.IOP_distnum, LPR.OS_distnum, and LPR.job_distnum respectively.
4. The next twelve consecutive 16-bit fields contain the following CUMSTATS in the indicated order:
 - (a) cumstat.RTX_100
 - (b) cumstat.RTX_400
 - (c) cumstat.RRX_GOOD_100
 - (d) cumstat.RRX_GOOD_400
 - (e) cumstat.RRX_BAD_100
 - (f) cumstat.RRX_BAD_400
 - (g) cumstat.RRX_PROP
 - (h) cumstat.RTX_PROP
 - (i) duplicate packet received counter
 - (j) total retransmission counter
 - (k) cumstat.HTX_total
 - (l) cumstat.HRX_total

Items (i) and (j) are the least significant 16 bits of the following CUMSTAT sums:

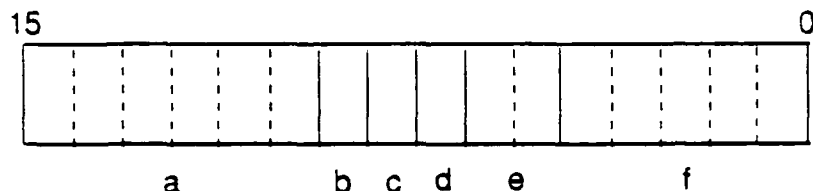
duplicate packet received counter

cumstat.RRX_FWD_DUP_RTXQ +
 cumstat.RRX_FWD_DUP_FILTER +
 cumstat.RRX_DID_DUP_RTXQ +
 cumstat.RRX_DID_DUP_FILTER +
 cumstat.RRX_DID_DEV_DUP_TXQ +
 cumstat.RRX_DID_DEV_DUP_FILTER

total retransmission counter

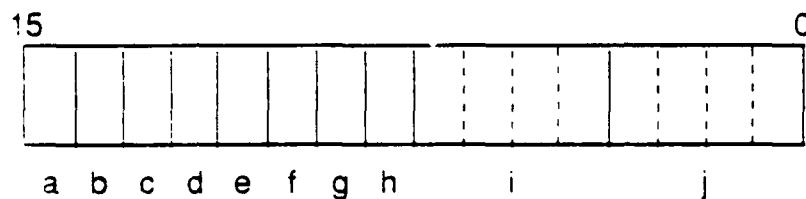
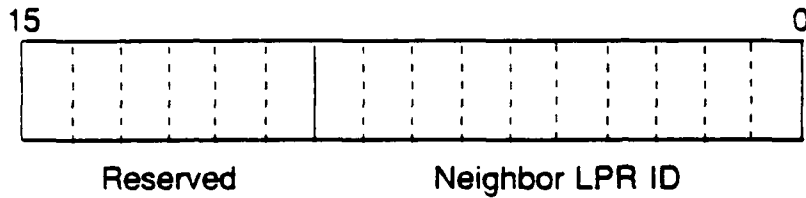
cumstat.RTX_NBR_OFF_TXC1 +
 cumstat.RTX_NBR_OFF_TXC2 +
 cumstat.RTX_NBR_OFF_TXC3 +
 cumstat.RTX_NBR_OFF_TXC4 +
 cumstat.RTX_NBR_OFF_TXC5 +
 cumstat.RTX_NON_OFF_TXC1 +
 cumstat.RTX_NON_OFF_TXC2 +
 cumstat.RTX_NBR_TOS_TXC1 +
 cumstat.RTX_NBR_TOS_TXC2 +
 cumstat.RTX_NON_TOS_TXC1 +
 cumstat.RTX_NON_TOS_TXC2

5. The next 32-bit field is the time the protocol came up in units of 6.656 milliseconds. It is right-shifted to be consistent with the timestamp in the MDP header.

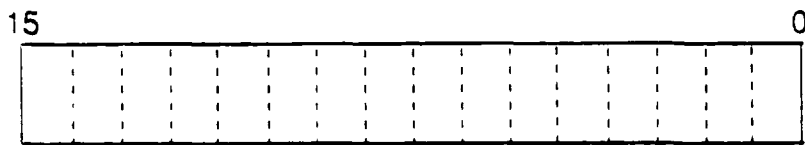


- (a) Time slot duration `LPR.time_slot_duration`
 - (b) `LPR.hdlc_dte_dce`, DTE = 1, DCE = 0
 - (c) `LPR.remote_cntrl_antr`, ON = 1, OFF = 0
 - (d) `LPR.auto_restart`, ON = 1, OFF = 0
 - (e) `LPR.RF_power` maximum transmitting power
 - (f) `LPR.rf_freq` LPR frequency setting
6. Two consecutive 16-bit fields: the smoothed average and deviation of the packet length (in words, one word equals two bytes) of all transmitted packets. See Appendix C for details.
 7. A consecutive sequence of 16-bit field pairs, one pair for each monitor; the first field contains the monitor's ID, the second contains the MDP transmission counter. The maximum number of monitors is 3. The monitor ID is the ID of one of the monitors that requested the MDP #1: `mdp1table.monitor_ID`. The MDP transmission counter is the number of MDP messages that have been sent to this monitor: `mdp1table.ID_TX_counter`.
 8. A consecutive sequence of 16-bit fields, one for each device. Each field contains the device ID.

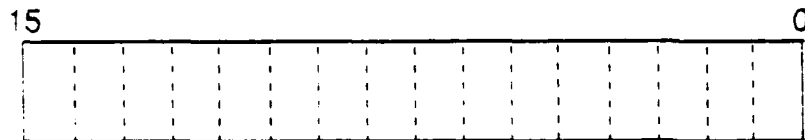
9. A consecutive sequence of 16-bit quintets, one for each neighbor. Each quintet has the following information:



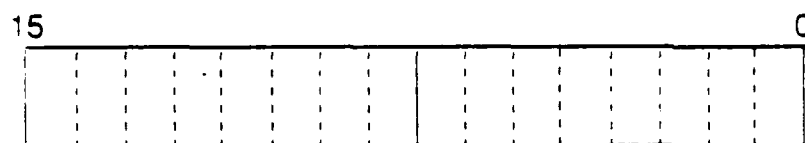
- (a) `nbr_table.good_neighbor`
- (b) `nbr_table.good_rx_100`
- (c) `nbr_table.good_rx_400`
- (d) `nbr_table.good_tx_100`
- (e) `nbr_table.good_tx_400`
- (f) Is this neighbor's frequency different from my frequency? 0 = false, 1 = true.
- (g) Is this neighbor's time slot different from my time slot? 0 = false, 1 = true.
- (h) Is this neighbor in time sync with me? 0 = false, 1 = true.
- (i) The link receive quality at 100 Kbps: `nbr_table.rx_Q_100`
- (j) The link receive quality at 400 Kbps: `nbr_table.rx_Q_400`



Transmit Counter to Nbr



Receive Counter from Nbr



LPR's AGC

Nbr's AGC

N. B. The last four counters are the least-significant 16 bits of the respective CUMSTATs:

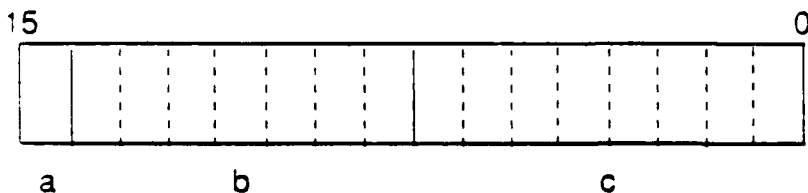
- `neighbor.table.cumstat_net_tx`
- `neighbor.table.cumstat_net_rx`
- `neighbor.table.RX_AGC` This is the AGC measurement for the last PROP from this neighbor.
- `neighbor.table.measured_AGC` This is the AGC measurement by the neighbor for the last PROP it received from this LPR, then reported in last PROP from this neighbor.

10. Trigger reports

For each Trigger there is a 16-bit trigger-counter field that is the least-significant 16 bits of `cumstat_ET`. The counters are in the same order as the triggers in Appendix B.

11. Hotlist Information

There is a single sixteen-bit field that contains the number of hotlisted packet radios. For each hotlisted packet radio there is a sixteen-bit field that gives its ID; its followed by the following sixteen-bit field:



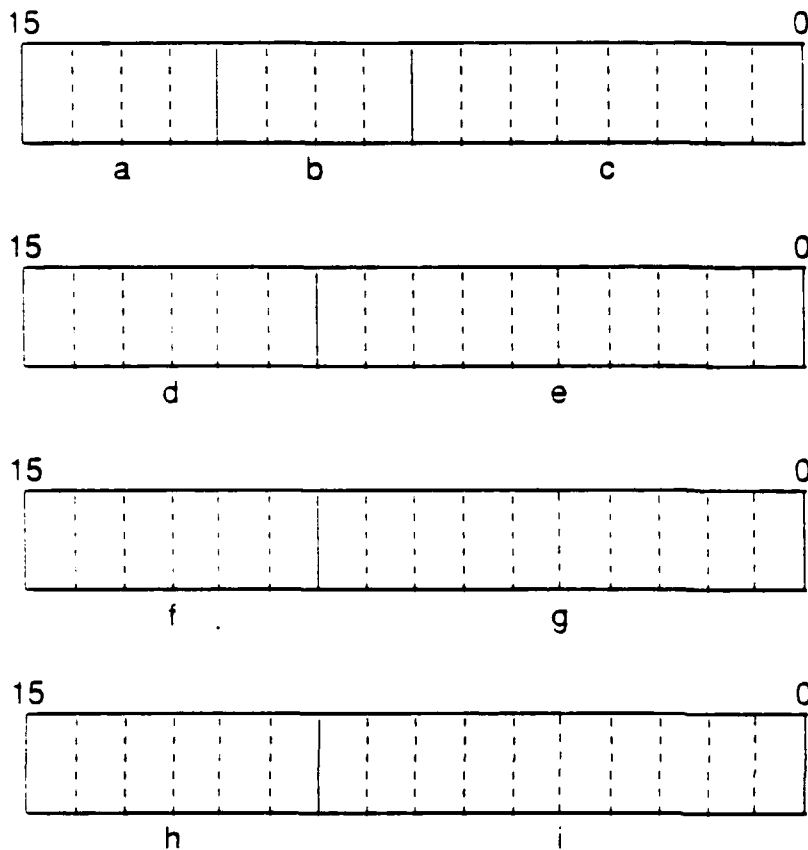
- (a) "1" if LPR was a neighbor when hotlist was received; "0" otherwise.
- (b) number of "Request for Authentication" packets received from hotlisted LPR
- (c) number of packets received from hotlisted LPR

A.3 MDP Type # 2 – Routing Information

This packet contains the following fields:

1. The packet header
 - (a) Version Number
 - (b) Reserved
 - (c) Number of LPR and Cluster entries in packet
 - (d) Reserved
 - (e) Previous PCN ID
 - (f) HRT sequence number
 - (g) Current PCN ID
 - (h) Reserved
 - (i) Next PCN ID
2. The following 2-byte fields are included for each LPR and each Cluster
 - (a) This bit is always set to "1"
 - (b) PCN bitmask from the tier entry
 - (c) Cluster-LPR flag. 0 = PR; 1 = Cluster
 - (d) Good-Bad Tier data flag. 0 = Good; 1 = Bad;
 - (e) Destination LPR or Cluster ID
 - (f) Distance in hops to the destination LPR
 - (g) Reporting LPR ID. (The least-significant 10 bits of the ID of the LPR through which to forward packets.)

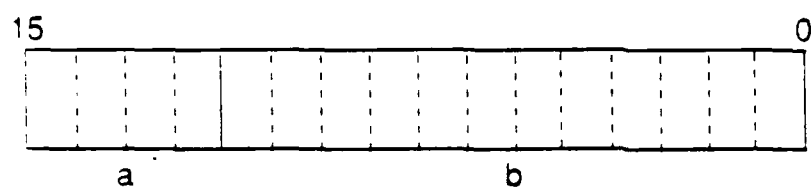
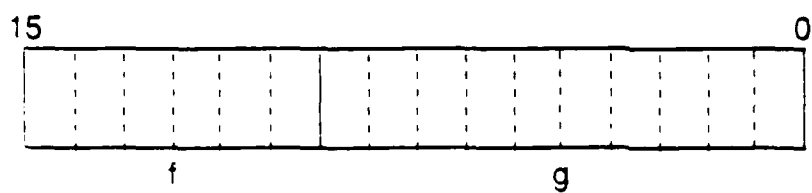
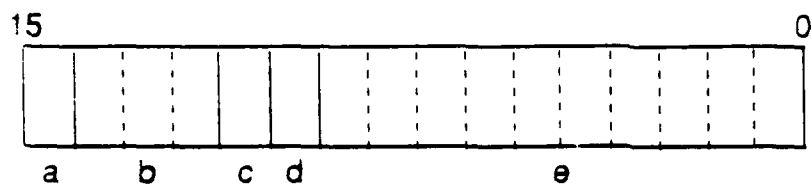
An additional 16-bit field contains the attached device ID for each device attached to this LPR. (The number can range between zero and eight, inclusive.)



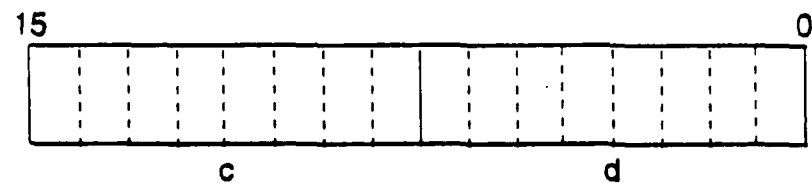
A.4 MDP Type # 3 - CUMSTAT Values

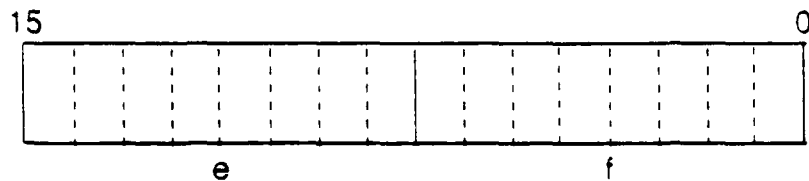
This packet contains LPR CUMSTATs. It contains the following fields:

1. The packet header
 - (a) MDP #3 Header Version
 - (b) Reserved
 - (c) Number of oldest version that can interpret the current packet format. (Later versions have just appended new counters.)
 - (d) Version number of current ordering of CUMSTATs.
 - (e) Number of regular CUMSTATs
 - (f) Number of Fault CUMSTATs
2. A 32-bit field giving the time since the CUMSTATs were last cleared. Time is measured in units of 6.656 milliseconds.



3. A 32-bit field for each *regular* CUMSTAT.
4. A 32-bit field for each *fault* CUMSTAT.

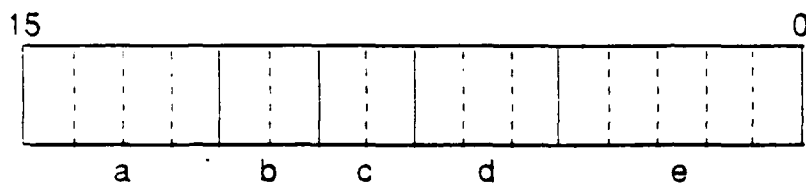




A.5 MDP Type # 4 - Neighbor LPR Data

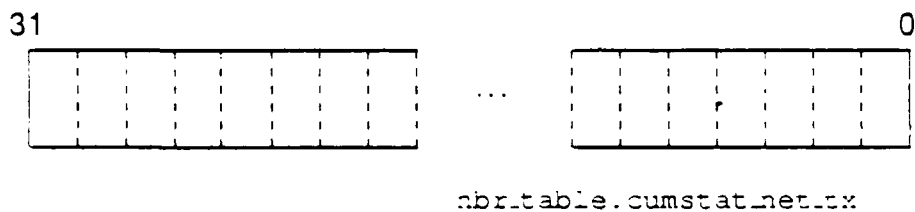
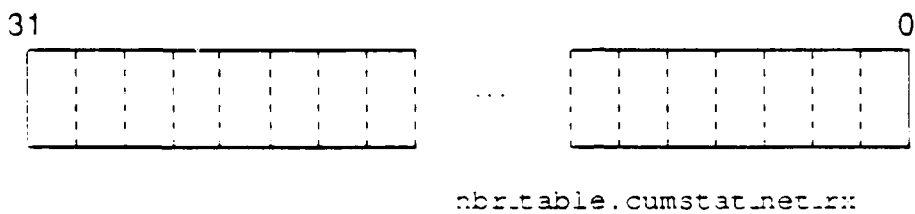
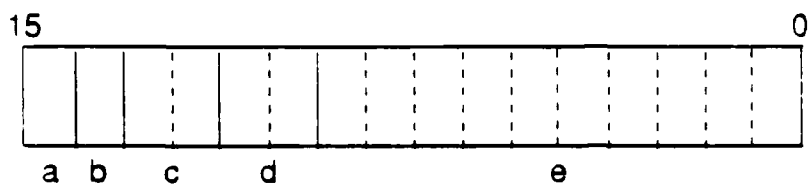
This packet contains LPR neighbor data. It has the following fields:

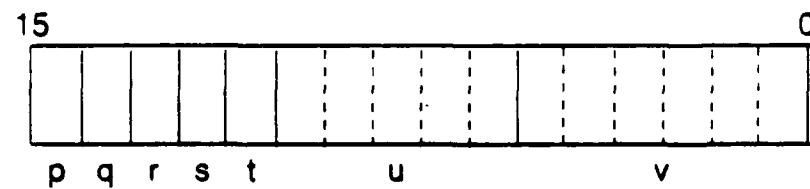
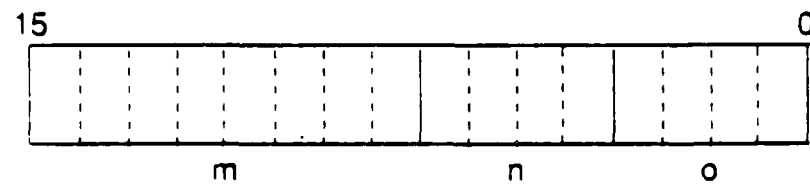
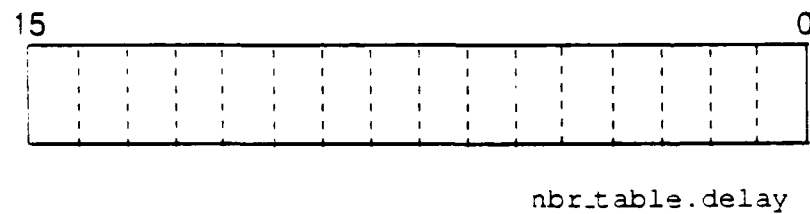
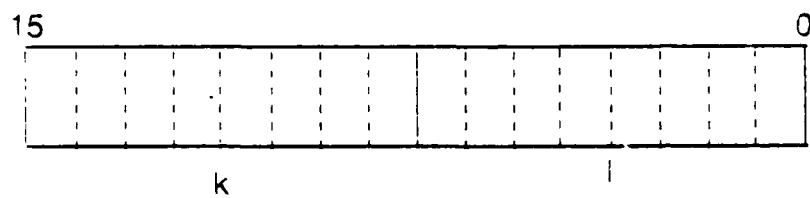
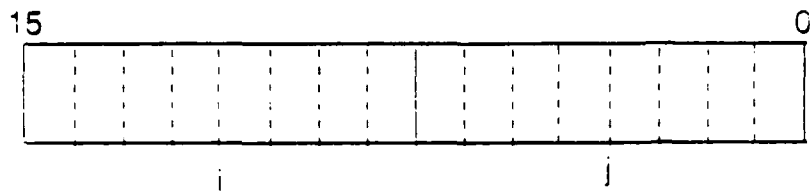
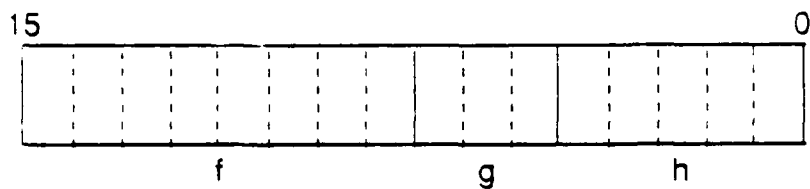
1. The packet header



- (a) Version Number
 - (b) Reserved
 - (c) Number of MDP #1 Monitors
 - (d) Reserved
 - (e) Number of Neighbors
2. A 32-bit field that gives the time since the CUMSTATs were last cleared. The units are 6.656 milliseconds.

3. A sequence of fields, shown below, are repeated for each neighbor:





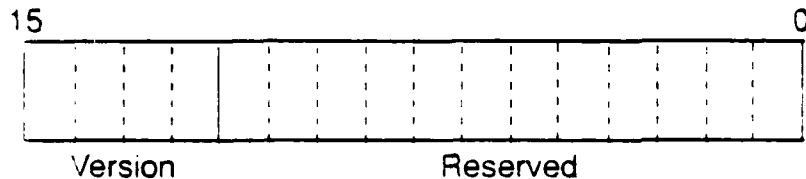
The fields are defined as follows:

- (a) Reserved
 - (b) Data rate on last PROP. 1 = 400 Kbps, 0 = 100 Kbps.
 - (c) Maximum power output by this neighbor. `nbrtable.RF.power`
 - (d) FEC rate on last received PROP. (The appropriate bit is taken from `nbrtable.preamble.data`.)
 - (e) Neighbor ID
 - (f) AGC value measured by neighbor on last PROP received from this LPR, then reported in last PROP from neighbor `neighbor.table.measured_AGC`.
 - (g) Reserved
 - (h) Neighbor frequency `nbrtable.freq_chnl`
 - (i) FEC error count `nbrtable.FEC.error.count`
 - (j) Multipath `nbrtable.multipath`
 - (k) Noise `nbrtable.noise`
 - (l) Signal + Noise Ratio from last received PROP `nbrtable.signal_plus_noise`
 - (m) AGC value measured by LPR on last PROP received from neighbor `nbrtable.rx_Q_100`
 - (n) Link quality at 100 Kbps `nbrtable.rx_Q_100`
 - (o) Link quality at 400 Kbps `nbrtable.rx_Q_400`
 - (p) `nbrtable.good_neighbor`
 - (q) `nbrtable.good_rx_100`
 - (r) `nbrtable.good_rx_400`
 - (s) `nbrtable.good_tx_100`
 - (t) `nbrtable.good_tx_400`
 - (u) `nbrtable.single_ended_sync_calc`
 - (v) `nbrtable.last_time_slot`
4. Two fields for each monitor: a 16-bit field that gives the monitor ID, and a 32-bit field that gives `mdp.table.cumstat_MDPs_TX`.

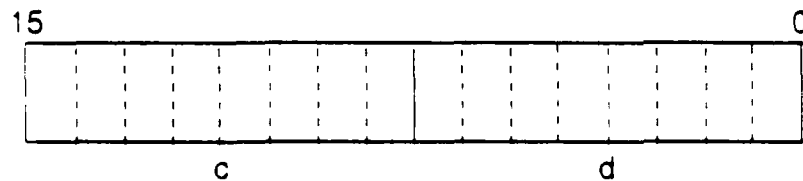
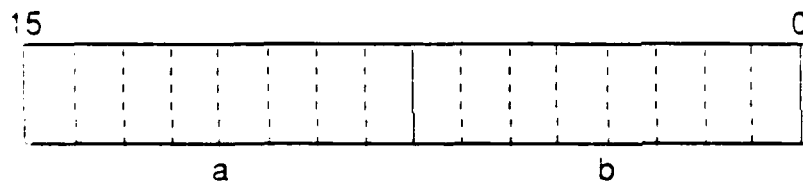
A.6 MDP Type # 5 - LPR Module Versions

This packet contains LPR module versions. It has the following fields:

1. The packet header



2. A 32-bit field that gives the IOP distribution number
3. A 32-bit field that gives the OS distribution number
4. A 32-bit field that gives the SURAP distribution number

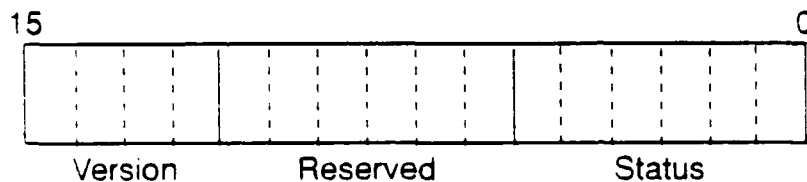


5. (a) Protocol string length in bytes (including ETX)
(b) OS string length in bytes
(c) IOP string length in bytes
(d) Reserved
6. A sequence of one-byte Protocol string character fields. The number of fields equals the protocol length.
7. A sequence of one-byte OS string character fields. The number of fields equals the OS length.
8. A sequence of one-byte IOP string character fields. The number of fields equals the IOP length.

A.7 MDP Type # 6 - LPR Fault Queue

This packet contains LPR fault Queue information. It has the following fields:

1. The packet header

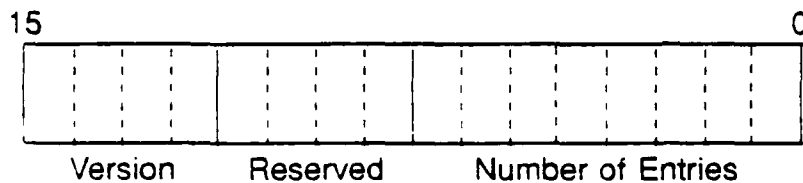


2. A 32-bit field that gives the time since the fault queue was last cleared. The units are 6.656 milliseconds.
3. The following sequence which is repeated for each fault:
 - a) A 16-bit field. The least significant eight bits give the number of occurrences; the most significant eight bits give the fault number. (The least significant eight bits of `countstat.this_fault`).
 - b) A 32-bit field that gives the fault timestamp in units of 6.656 milliseconds.
 - c) A sequence of 16-bit fault data fields. The number of fields is equal to the fault length (currently 3).

A.8 MDP Type # 7 - LPR Timer Values

This packet contains LPR timing values. It contains the following fields:

1. The packet header

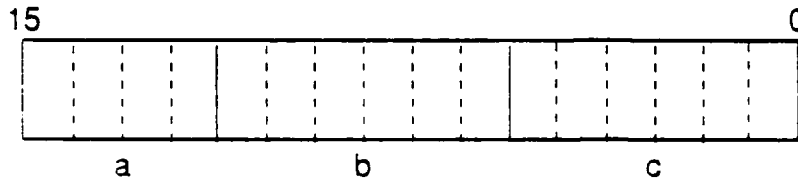


2. A 32-bit field that gives the time since the CUMSTATs were last cleared. The units are 6.656 milliseconds.
3. a 16-bit field. The least significant eight bits are the current version number; the most-significant eight bits are the oldest useful version. The current version number is needed to determine the order of the timers. The oldest useful version is the oldest version that can interpret this packet format. (Later versions have just appended new timers.)
4. The following sequence which is repeated for each entry:
 - (a) A 16-bit field containing the minimum timer value
 - (b) A 16-bit field containing the maximum timer value
 - (c) A 16-bit smoothed average timer value

A.9 MDP Type # 8 - LPR Command NACK

This packet contains an LPR Command NACK. It contains the following fields:

1. The packet header



- (a) Version
- (b) sub_type of MDP #0 that cannot be transmitted. See Section 1.
- (c) NACK reason:
 - 0 = MDP_IGNORED_SNAPSHOT The MDP request was ignored because the snapshot buffer was in use. No Data included.
 - 1 = MDP_IGNORED_PKT_IN_USE The MDP request was ignored because the LPR has an outstanding MDP in the network.
 - 2 = CANT_SET_MDP_DEST Cannot satisfy the MDP_ME_MDP1 request. In this case three 16-bit fields will follow that contain the LPR's current monitor IDs.
2. Three 16-bit fields that contain data when the NACK reason is #2; otherwise these fields are not included.

Appendix B. MDP Triggers

This Appendix is a complete list of all MDP triggers.

- 0 = MAX_TIMEOUT Send an MDP #1 every thirty minutes.
- 1 = PR_FAULT Send an MDP #1 each time an internally detected LPR fault occurs.
- 2 = NBR_STAT_CHNG Send an MDP #1 each time a neighbor status change occurs, e.g., good to bad, bad to good, bad to erased.
- 3 = DEV_CHNG Send an MDP #1 whenever the device list changes.
- 4 = PARAM_CHANGE Send an MDP #1 every time one of the following parameters changes:
 - 1. frequency
 - 2. maximum transmission power used for all neighbors
 - 3. time slot duration.

Appendix C. Algorithms for Packet-Length Statistics

We use the following notation:

$t = 0, 1, 2, \dots$ The sequence of measurement indices.

A_t = the current mean packet length.

P_t = the current packet length measurement

V_t = the current packet length mean deviation estimate.

The estimates are computed by the following formulas:

$$A_t = \frac{7}{8} A_{t-1} + \frac{1}{8} P_t \quad (C.1)$$

$$V_t = \frac{7}{8} V_{t-1} + \frac{1}{8} |P_t - A_t| \quad (C.2)$$